智能系统学报
CAAI TRANSACTIONS ON INTELLIGENT SYSTEMS

基于隐式随机梯度下降优化的联邦学习

窦勇敢, 袁晓彤

引用本文:
窦勇敢,袁晓彤. 基于隐式随机梯度下降优化的联邦学习[J]. 智能系统学报, 2022, 17(3): 488–495.
DOU Yonggan,YUAN Xiaotong. Federated learning with implicit stochastic gradient descent optimization[J]. *CAAI Transactions on Intelligent Systems*, 2022, 17(3): 488–495.

在线阅读 View online: https://dx.doi.org/10.11992/tis.202106029

DOI: 10.11992/tis.202106029
网络出版地址: https://kns.cnki.net/kcms/detail/23.1538.TP.20211213.1548.002.html

基于隐式随机梯度下降优化的联邦学习

窦勇敢[1,2], 袁晓彤[1,2]
（1. 南京信息工程大学 自动化学院, 江苏 南京 210044; 2. 江苏省大数据分析技术重点实验室, 江苏 南京 210044）

摘　要: 联邦学习是一种分布式机器学习范式, 中央服务器通过协作大量远程设备训练一个最优的全局模型。目前联邦学习主要存在系统异构性和数据异构性这两个关键挑战。本文主要针对异构性导致的全局模型收敛慢甚至无法收敛的问题, 提出基于隐式随机梯度下降优化的联邦学习算法。与传统联邦学习更新方式不同, 本文利用本地上传的模型参数近似求出平均全局梯度, 同时避免求解一阶导数, 通过梯度下降来更新全局模型参数, 使全局模型能够在较少的通信轮数下达到更快更稳定的收敛结果。在实验中, 模拟了不同等级的异构环境, 本文提出的算法比 FedProx 和 FedAvg 均表现出更快更稳定的收敛结果。在相同收敛结果的前提下, 本文的方法在高度异构的合成数据集上比 FedProx 通信轮数减少近 50%, 显著提升了联邦学习的稳定性和鲁棒性。

关键词: 联邦学习; 分布式机器学习; 中央服务器; 全局模型; 隐式随机梯度下降; 数据异构; 系统异构; 优化算法; 快速收敛
中图分类号: TP8　　**文献标志码**: A　　**文章编号**: 1673−4785(2022)03−0488−08

中文引用格式: 窦勇敢, 袁晓彤. 基于隐式随机梯度下降优化的联邦学习 [J]. 智能系统学报, 2022, 17(3): 488–495.
英文引用格式: DOU Yonggan, YUAN Xiaotong. Federated learning with implicit stochastic gradient descent optimization[J]. CAAI transactions on intelligent systems, 2022, 17(3): 488–495.

Federated learning with implicit stochastic gradient descent optimization

DOU Yonggan[1,2], YUAN Xiaotong[1,2]
(1. School of Automation, Nanjing University of Information Science and Technology, Nanjing 210044, China; 2. Jiangsu Key Laboratory of Big Data Analysis Technology, Nanjing 210044, China)

Abstract: Federated learning is a distributed machine learning paradigm. The central server trains an optimal global model by collaborating with numerous remote devices. Presently, there are two key challenges faced by federated learning: system and statistical heterogeneities. Herein, we mainly focus on the slow convergence of the global model or when it even fails to converge due to system and statistical heterogeneities. We propose a federated learning optimization algorithm based on implicit stochastic gradient descent optimization, which is different from the traditional method of updating in federated learning. We use the locally uploaded model parameters to approximate the average global gradient and to avoid solving the first-order and update the global model parameter via gradient descent. This is performed so that the global model can achieve faster and more stable convergence results with fewer communication rounds. In the experiment, different levels of heterogeneous settings were simulated. The proposed algorithm shows considerably faster and more stable convergence behavior than FedAvg and FedProx. In the premise of the same convergence results, the experimental results show that the proposed method reduces the number of communication rounds by approximately 50% compared with Fedprox in highly heterogeneous synthetic datasets. This considerably improves the stability and robustness of federated learning.

Keywords: federated learning; distributed machine learning; central server; global model; implicit stochastic gradient descent; statistical heterogeneity; systems heterogeneity; optimization algorithm; faster convergence

收稿日期: 2021−06−18.　**网络出版日期**: 2021−12−14.
基金项目: 国家自然科学基金项目（61876090,61936005）; 科技创新 2030–“新一代人工智能”重大项目（2018AAA0100400）.
通信作者: 袁晓彤. E-mail: xtyuan1980@gmail.com.

近些年来, 随着深度学习的兴起, 人们看到了人工智能的巨大潜力, 同时希望人工智能技术应用到更复杂和尖端的领域。而现实状况是数据分

散在各个用户或行业中, 用户数据存在隐私上的敏感性和安全性。如何在保护数据隐私前提下进行机器学习模型训练, 让人工智能技术发挥出更强大的作用成为一种挑战。

为了让这些隐私数据流动起来, 同时应对非独立同分布数据的影响, Google 科学家 Mcmahan 等[1] 提出联邦学习 (federated learning), 通过协调大量远程分布式设备在保护用户数据隐私的前提下训练一个高质量的全局模型。

目前的联邦学习算法还存在诸多问题。首先, 每个设备 CPU、GPU、ISP、电池以及网络连接 (3G、4G、5G、WIFI)[2] 等硬件差异导致设备间存在很大的系统异构性。传统的联邦学习方法 FedAvg[1] 在规定时间内将没有训练结束的设备简单丢弃, 这在现实情况中是不可取的, 浪费了大量的计算资源。其次, 每个设备的数据分布和类型存在很大的差异[3], 跨设备的数据是非独立同分布的(non-IID), 这是数据的异构性。不同的异构环境中模型的收敛效果差别很大, 甚至无法收敛。这些系统级别的异构性给联邦学习带来了极大的挑战。

现有针对异构性问题的分布式优化算法中, 大部分都是针对特定异构环境设定的。例如:文献 [4-6] 提出让所有设备都参与每一轮的训练, 虽然在异构数据环境中的收敛性得到了保证, 但是这在现实的联邦环境[1] 中是不可行的。这不仅增加了服务器的通信负担, 而且参与联邦训练的设备也应随机抽取。也有方法通过共享本地数据来解决数据异构性的问题[7-8], 但这违背了联邦学习保护用户数据隐私的前提。在联邦设置中, 文献 [9] 通过在服务器端设计基于动量优化器 FEDYOGI 来加快异构数据环境中全局模型收敛速度, 这虽然提高了模型的收敛速度, 但却增加了服务器的计算量, 在有限的计算资源下不是好的选择。此外, 也有研究者利用二阶拟牛顿法优化模型[10], 在相同的异构环境中, 与 FedAvg 相比达到相同精度下减少了通信轮数, 提高了通信效率, 但这潜在增加了客户端本地的计算量。

除了数据异构性, 每个参与联邦训练的客户端的硬件存在差异, 这导致设备间存在很大的系统异构性[11]。例如: 在文献 [12-15] 中, 介绍了在异构环境中目前最新的联邦学习研究进展, 在全局模型聚合阶段的更新方式同 FedAvg[1] 一样, 在指定的时间窗口内, 服务器将未完成训练的设备直接丢弃, 不允许上传本轮训练的模型参数。各参与训练的设备不能根据自己硬件性能在本地执行可变数量的本地工作, 缺乏自主调节能力。

在解决联邦学习异构性的问题上, 近邻优化的更新方式广泛地用于研究, 包括高效通信分布式机器学习[16]、联邦学习中公平性和鲁棒性的权衡[17]。近邻优化在原理上与有偏正则化相同, 其中文献 [18] 中考虑有偏正则化的方法对 FedAvg 进行重新参数化, 提出 FedProx, 通过有偏正则化约束每个设备学习的本地模型更加接近于全局模型, 并允许各参与训练的设备在本地执行可变数量的工作, 在异构环境中提供了收敛的保证。由于 FedProx 在优化全局模型参数w时和 FedAvg 方式相同, 通过简单平均本地上传的模型参数来更新全局模型参数, 导致全局模型收敛速度慢, 缺乏直接对全局模型参数的优化。

受小批量近似更新的元学习机制[19] 的启发, 本文提出了基于隐式随机梯度下降优化的联邦学习算法, 在本地模型更新阶段通过近邻优化约束本地模型更新更加接近于全局模型, 在全局模型聚合阶段通过求解近似全局梯度, 利用梯度下降来更新全局模型参数。最终实现全局模型能够在较少的通信轮数下达到更快更稳定的收敛结果。

本文的贡献主要体现在以下 3 个方面:

1) 区别于已有的方法, 不在对全局模型参数进行简单平均。在全局模型聚合阶段, 通过利用本地上传的模型参数近似求出平均全局梯度, 同时也避免求解一阶导数。

2) 针对异构性导致的全局模型收敛慢甚至无法收敛的问题, 区别于现有的联邦学习算法, 本文提出基于隐式随机梯度下降优化的联邦学习算法, 通过隐式随机梯度下降来更新全局模型参数, 能够使全局模型参数实现更加高效的更新, 从而可以在有限的计算资源下加快模型的收敛速度。

3) 和现有的工作相比, 本文的算法在高度异构的合成数据集上, 30 轮左右就可以达到 FedAvg 的收敛效果, 40 轮左右可以达到 FedProx 的收敛效果。在相同收敛效果的前提下, 本文的算法比 FedProx 减少了近 50% 的通信轮数。

1　客户端–服务器的联邦学习更新架构

联邦学习更新架构主要有客户端–服务器和去中心化对等计算架构。其中最常用的是客户端–服务器的联邦学习更新架构。训练过程主要分为两个阶段: 本地模型更新阶段和全局模型聚合阶段。具体更新过程如图 1 所示。

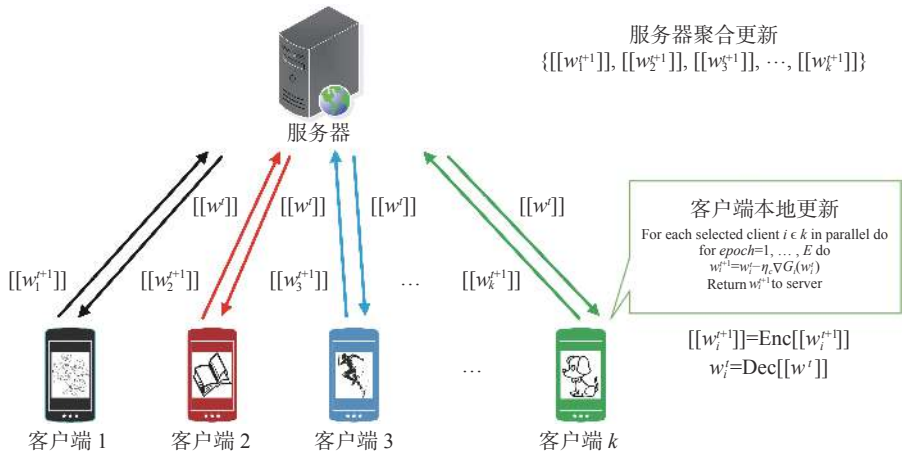


图 1　客户端-服务器联邦学习架构

Fig. 1　Federated learning architecture of client and server

1)本地模型更新

在本地模型更新阶段，服务器首先随机选取K个客户端，然后服务器发送全局模型参数$[[w^t]]$给被选客户端，客户端利用本地数据并行执行E个 epoch 的随机梯度下降，然后将更新后的模型参数经过同态加密算法[20]加密$[[w_i^{t+1}]]=\mathrm{Enc}[w_i^{t+1}]$，之后再上传至服务器。

2)全局模型聚合

服务器聚合来自各个客户端加密后的模型参数$\{[[w_1^{t+1}]],[[w_2^{t+1}]],[[w_3^{t+1}]],\cdots,[[w_k^{t+1}]]\}$，对模型参数进行加权平均计算，最后服务器将更新后的全局模型参数$[[w^t]]$发送给下一轮被选客户端。客户端接收全局模型参数并将其解密$w_i^t=\mathrm{Dec}\,[[w^t]]$，进行下一轮模型参数更新。重复上述过程，直至损失收敛。

2　隐式随机梯度下降优化的联邦学习算法设计

在本节中，主要介绍联邦近邻优化算法和隐式随机梯度下降优化算法的关键要素。由于联邦学习是通过大量设备与中央服务器协同学习一个最优的全局模型，因此我们的最终目标是最小化：

$$\min_{w}\mathbf{E}_k\left[\min_{w_k}\left\{F_k(w_k)+\frac{\lambda}{2}\|w_k-w\|_2^2\right\}\right] \tag{1}$$

式中：w_k是设备k在本地迭代过程中所得的近似最优解；w是需要求解全局模型的最优解；$F_k(w_k):=\mathbf{E}_{x_k\sim\mathcal{D}_k}[\mathcal{L}(f(w_k,x_i),y_i)]$，每个设备本地数据$x_k$服从不同的分布$\mathcal{D}_k$，损失函数是预测值与真实值之间的差。式 (1) 包含两方面的优化过程：1)在本地模型训练阶段，每个设备通过全局模型参数w学习一个本地近似最优w_k；2)在全局模型聚合阶段，服务器通过各设备上传的w_k利用隐式随机梯度下降来调整全局模型参数w，使w与所有w_k的平均距离较小。具体的算法流程为：

输入　w^0随机初始化参数，N总设备；

输出　最终的全局模型参数w^{t+1}。

1) FOR 全局轮数$t=0,1,\cdots,T-1$;

2) Server 以概率p_k随机选取K个设备并指定固定学习率η_c;

3) Server 发送当前全局模型w^t给所选设备;

4) 每个设备$k=1,2,\cdots,K$并行计算：$F_k(w_k)=\frac{1}{n_k}\sum_{i\in n_k}\mathcal{L}(f(w_k,x_i),y_i)$;

5) $w_k^{t+1}=\arg\min_{w_k}\left[F_k(w_k)+\frac{\lambda}{2}\|w_k-w^t\|_2^2\right]$;

6) 重复 4)~ 5)，结束并行计算，每个设备将计算结果w_k^{t+1}传至 Server;

7) Server 以固定轮数衰减的学习率η_g对$\lambda(w^t-w_k^*)$做梯度下降更新全局模型参数：

$$w^{t+1}=w^t-\eta_g\lambda\left(w^t-\frac{1}{K}\sum_{k\in S_t}w_k^{t+1}\right);$$

8) Server 将更新后的模型参数w^{t+1}发送给 Clients;

9) 重复 2)~8) t次；

10) END

在算法 1 中，步骤 4)~6) 为本地模型训练阶段，7)~9) 为 Server 全局模型更新阶段，然后将更新后的模型参数发送给下一轮参与训练的设备。不断重复以上过程，直至模型损失收敛。

2.1　联邦近邻优化

在本地模型训练阶段，主要在本地模型更新时引入带参数的近邻算子约束本地模型更新更加接近于全局模型，这种本地优化算法被称为 FedProx 算法[18]，每个设备k的本地目标函数被重新定义为

$$\min_{w_k}G_k(w_k)=F_k(w_k)+\frac{\lambda}{2}\left\|w_k-w^t\right\|_2^2 \tag{2}$$

式中: λ是一个约束本地模型和全局模型差异的超参数; w^t表示在第t轮服务器聚合更新之后的全局模型参数。

2.2　基于隐式随机梯度下降的全局模型更新优化

本地训练结束后, 每个设备将更新后的模型参数w_k^{t+1}上传至服务器, 服务器通过聚合本地模型参数更新全局模型参数。从加快全局模型收敛速度的目标出发, 在服务器全局模型聚合阶段利用隐式随机梯度下降算法对全局模型参数进一步优化, 使其能够在有限的资源下以更少的通信轮数达到更快更稳定的收敛效果。

由于客户端本地目标函数为$\min\limits_{w_k} G_k(w_k)=F_k(w_k)+\dfrac{\lambda}{2}\|w_k-w^t\|_2^2$, 假设$w_k^*$是目标函数$G_k(w_k)$的最优解, 且函数$F_k(w_k)$可微。由一阶最优条件可以得到:

$$\nabla F(\boldsymbol{w}_k^*)+\lambda(\boldsymbol{w}_k^*-\boldsymbol{w}^t)=0 \tag{3}$$

由链式法则可以得到:

$$\begin{aligned}\nabla G_k(\boldsymbol{w}^t)&=\left(\frac{\partial \boldsymbol{w}_k^*}{\partial \boldsymbol{w}^t}\right)^{\mathrm{T}}\nabla F_k(\boldsymbol{w}_k^*)+\\&\lambda\left(\boldsymbol{I}-\left(\frac{\partial \boldsymbol{w}_k^*}{\partial \boldsymbol{w}^t}\right)^{\mathrm{T}}\right)(\boldsymbol{w}^t-\boldsymbol{w}_k^*)=\\&\lambda(\boldsymbol{w}^t-\boldsymbol{w}_k^*)+\left(\frac{\partial \boldsymbol{w}_k^*}{\partial \boldsymbol{w}^t}\right)^{\mathrm{T}}*(\nabla F_k(\boldsymbol{w}_k^*)+\lambda(\boldsymbol{w}_k^*-\boldsymbol{w}^t))=\\&\lambda(\boldsymbol{w}^t-\boldsymbol{w}_k^*)\end{aligned} \tag{4}$$

所以$\nabla G_k(\boldsymbol{w}^t)=\lambda(\boldsymbol{w}^t-\boldsymbol{w}_k^*)$, 式(4)展现了全局模型的梯度估计可以通过求解当前任务的近似更新来计算。在第t轮, 所选设备在本地数据集上利用随机梯度下降更新 E 轮后, 求出近似最优解w_k^{t+1}。服务器通过式 (4) 可以计算出平均的全局梯度:

$$\nabla G(w^t)=\lambda\left(w^t-\frac{1}{K}\sum_{k\in S_t}w_k^*\right) \tag{5}$$

由于精确的本地最优解w_k^*很难去估计, 本文用次优解w_k^{t+1}来代替w_k^*, 因此全局模型参数更新为

$$w^{t+1}=w^t-\eta_g\lambda\left(w^t-\frac{1}{K}\sum_{k\in S_t}w_k^{t+1}\right) \tag{6}$$

式中: S_t为 K 个设备的子集; t为当前训练轮数; $\eta_g=\dfrac{1}{t}\eta_{gi}$为按固定轮数衰减的学习率; η_{gi}为初始化学习率, 在训练模型初期用较大的学习率对全局模型进行优化, 随着通信轮数的不断增加学习率逐步减小, 有效保证了全局模型在训练过程中能以较快的速度逐步趋于稳定。更新后的w^{t+1}作为下一轮训练的全局模型参数。

从式 (3)~(6) 推导过程很容易看出, 本文提出基于隐式随机梯度下降优化的联邦学习算法是直接对全局模型参数进行优化, 而不是简单平均所有设备上传的本地模型参数作为更新后的全局模型参数。因为$\nabla G_k(w^t)=\lambda(w^t-w_k^*)$, 所以在服务器端只需通过$\lambda\left(w^t-\dfrac{1}{K}\sum\limits_{k\in S_t}w_k^*\right)$就可以得到平均全局模型梯度, 因此避免了求解一阶导数, 然后利用随机梯度下降对全局模型参数进行更新。相比于 FedProx, 本算法在信息比较冗余的情况下能更高效地利用有效信息。其次, 在迭代的过程中也会很快收敛到最小值附近, 加快模型的收敛速度。

3　实验与结果

为了验证本文提出的隐式随机梯度下降优化算法的有效性, 本文在 3 个真实数据集和 3 个合成数据集上进行实验, 在分类和回归任务上进行评估, 并与当前具有代表性的解决异构性问题的方法 FedProx[18] 以及经典的 FedAvg[1] 算法进行比较。

3.1　实验设置

在 Linux 系统下, 包括 2 块 GeForce GTX 1 080 Ti 和 1 块 GeForce GTX TITAN X 的服务器上进行仿真实验, 代码使用 Tensorflow 框架实现, 基于 Python3 来实现基于隐式随机梯度下降优化的联邦学习算法。其中, 训练轮数、每轮迭代次数、选择设备数量、学习率等超参数设置如表 1 所示。

表 1　超参数设置

Table 1　Setting of Hyperparameters

超参数	Synthetic	Snet140	MNIST	EMNIST
$N_{\text{num\_rounds}}$	200	800	200	200
$N_{\text{num\_epochs}}$	20	20	20	20
$B_{\text{batch\_size}}$	10	10	10	10
$N_{\text{num\_clients}}$	10	10	10	10
η_c	0.01	0.6	0.03	0.003
η_{gi}	0.75	1.1	0.75	0.95

为了保证评估方法与结果的公平性, 本文提出的方法与 FedProx、FedAvg 使用了相同的本地求解器, 在模拟系统异构设置时, 掉队的设备数量分别设置为 0%、50%、90%。生成合成数据集本文使用了和 FedProx 类似的方法, 通过式(7)生成本地数据:

$$y=\operatorname{argmax}(\operatorname{softmax}(Wx+b)) \tag{7}$$

式中: $W\in 10\times 60$; $x\in 60$; $b\in 10$。通过式(7)生成 30 个设备的数据集, 同样每轮随机抽取 10 个参与训练。

3.2　3 个真实数据集和模型

Sent140[21] 是一个 Twitter 带有表情的文本信息情感分类数据集，该任务使用的是一个两层 LSTM，包含 256 个隐藏层单元，每个 Twitter 帐户对应一个设备。该模型以 25 个字符序列作为输入，通过两个 LSTM 层和一个全连接层，每个训练样本输出一个字符。

MNIST[22] 是一个 0~9 手写体数字识别数据集，在这个任务上利用逻辑回归的方法研究手写数字图像分类问题。为了生成非独立同分布数据，本文将数据随机分布在 1 000 个设备中，每个设备只有 2 种数字。模型的输入是 28×28 维的图像，输出是 0~9 这 10 个数字的标签。

EMNIST[23] 是 MNIST 数据集的扩展，包含 0~9 数字和 26 个英文字母的大小写，构成了更大难度的 62 类手写字符图像分类任务，但在实验中只随机抽取 10 个小写字母，每个设备分配 5 个类，在这个任务上利用逻辑回归的方法研究图像分类问题。模型的输入是 28×28 维的图像，输出是 a~j 这 10 个类的标签。

对于以上所有数据集，客户端的本地数据分配遵循幂律分布[24]。本文在本地分配 80% 为训练集，20% 为测试集。各设备数据集组成如表 2 所示。

表 2　设备数据集分布

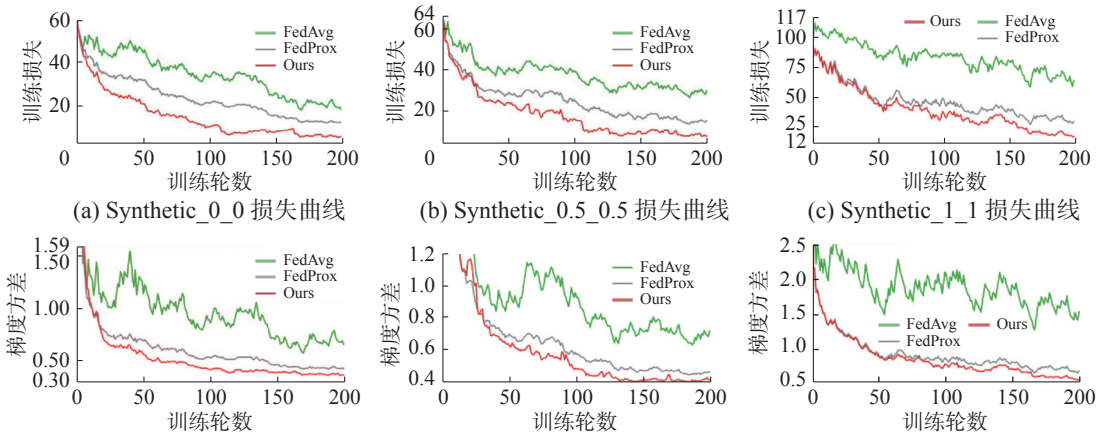
Table 2　Datasets distribution on devices

数据集	设备/台	总样本/个	平均样本/个
Sent140	772	40 783	53
MNIST	1 000	69 035	69
EMNIST	200	18 345	92

3.3　合成数据集实验结果分析

首先在第 1 个实验中，为了验证本文的算法在异构数据集上有更快的收敛速度，本文在 3 组合成数据集上进行实验，分别是 Synthetic_0_0、Synthetic_0.5_0.5、Synthetic_1_1，从左到右数据异构性逐渐增强，异构性越强，对模型收敛影响越大。本文通过损失的减小速度和梯度方差[25] 的变化来衡量模型的收敛速度，结果如图 2 所示。为了证明本文方法的公平性和有效性，约束项 λ 统一设置成相同的值。由图 2 训练损失和梯度方差可以看出，本文的方法在第 30 轮左右达到了 FedAvg 的收敛效果，在第 40 轮左右达到了 FedProx 的收敛效果，并且 40 轮以后还在继续收敛。梯度方差（variance of local gradient, VLG）越小表示越稳定，收敛性越好。VLG 可表示为

$$\mathrm{VLG}=\frac{1}{K}\sum_{k=1}^{K}\left\|\nabla G(w^{t})-\nabla G_{k}\left(w_{k}^{t+1}\right)\right\|^{2} \tag{8}$$



(a) Synthetic_0_0 损失曲线　(b) Synthetic_0.5_0.5 损失曲线　(c) Synthetic_1_1 损失曲线

(d) Synthetic_0_0 梯度方差曲线　(e) Synthetic_0.5_0.5 梯度方差曲线　(f) Synthetic_1_1 梯度方差曲线

图 2　合成数据集实验结果分析

Fig. 2　Analysis of experimental results of synthetic datasets

实验中，通过使所有设备执行相同的工作量来模拟不存在系统异构性的情况，随着数据异构性增强，全局模型收敛结果最终会趋于某个区间，因此本文取最后一半通信轮数的平均测试精度作为模型好坏的评判标准，在合成数据集上平均测试精度如表 3 所示，可以看出本文提出的算法平均测试精度普遍高于 FedProx 和 FedAvg。

表 3　合成数据集上平均测试精度

Table 3　Average test accuracy on synthetic datasets　%

数据集	FedAvg	FedProx	本文
Synthetic_0_0	79.6	83.6	**85.0**
Synthetic_0.5_0.5	79.3	81.7	**84.5**
Synthetic_1_1	69.7	75.6	**76.3**

3.4　真实数据集实验结果分析

在本实验中, 为了验证本文提出的算法在高度系统异构性和数据异构性环境下的整体效果, 本节在 3 个联邦学习常用真实数据集和一个合成数据集上比较不同算法的稳定性和收敛效果, 其中 Synthetic_1_1 客户端本地类别设置为 5, 实现在数据异构性基础上模拟不同系统异构性的联邦设置。

本文通过约束设备的本地工作量, 使每个设备训练指定的 E 来模拟系统的异构性, 对于不同的异构设置, 随机选择不同的 $E(E<20)$ 分配给 0%、50% 和 90% 当前参与训练的设备。当掉队者为 0% 时, 代表所有设备执行相同的工作量 (E=20)。在指定的全局时间周期内, 当 E<20 时, FedAvg 会丢掉这些掉队者, 本文的算法和 FedProx 会合并这些掉队者, 不同的是本文在全局模型聚合阶段会有效地使用合并掉队者的模型参数, 利用隐式随机梯度下降对全局模型进一步优化。真实数据集上的训练损失如图 3 所示, 从上到下 3 行图片分别代表 0%、50% 和 90% 的掉队者。随着迭代轮数的不断增加, 平均损失逐渐趋于稳定, 从图 3 中可以看出本文提出的算法的收敛速度明显优于 Fedavg 和 FedProx。

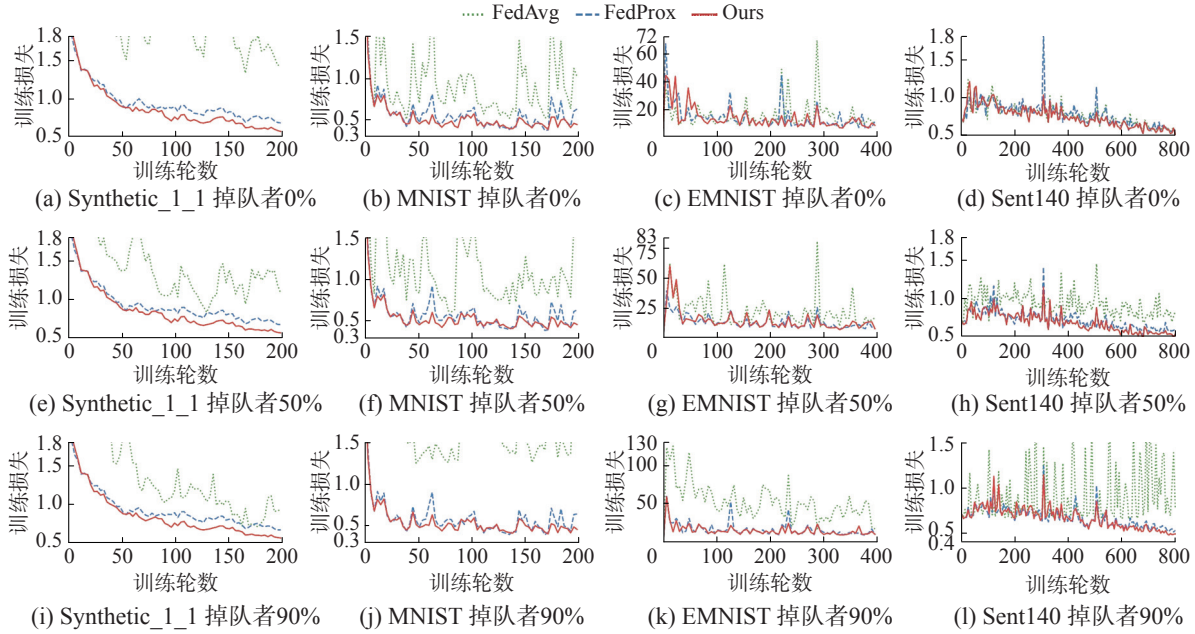


图 3　真实联邦数据集实验结果分析

Fig. 3　Analysis of experimental results of realistic federated datasets

表 4 给出了在高度异构环境下模型的平均测试精度, 从表中可以看出掉队者为 90% 时, 本文提出的算法的平均测试精度最高, 其次是 FedProx。本文算法在 MNIST 数据集上比 FedProx 高 5%。实验中, 在 Sent140 数据集上通过设置相同超参数进行比较不同算法运行时间, 在通信轮数为 200 的情况下, FedAvg、FedProx 和本文所提算法运行时间分别为 67 min、108 min、108 min。

表 4　高度异构环境各算法平均测试精度

Table 4　Average test accuracy of each algorithm in highlyheterogeneous environment　%

数据集	FedAvg	FedProx	本文算法
Synthetic_1_1	72.3	76.1	**77.4**
MNIST	76.7	81.4	**86.4**
EMNIST	50.4	68.1	**68.3**
Sent140	57.0	69.4	**69.5**

4　结束语

本文提出了一种基于隐式随机梯度下降优化的联邦学习算法。全局模型聚合阶段不再是简单的平均各设备上传的模型参数, 而是利用本地上传的模型参数近似求出全局梯度, 同时避免求解一阶导数。利用随机梯度下降对全局模型参数进行更新, 在信息冗余的情况下能更准确地利用有效信息, 随着通信轮数不断增加, 全局模型会很快收敛到最小值附近。在 3 个合成数据集和 3 个真实数据集上的实验结果充分表明: 该算法能够在不同异构环境中均表现出更快更稳健的收敛结果, 显著提高了联邦学习在实际应用系统中的稳定性和鲁棒性。

参考文献:

[1]　MCMAHAN B, MOORE E, RAMAGE D, et al. Commu-

nication-efficient learning of deep networks from decentralized data[C]//Proceedings of the 20th International Conference on Artificial Intelligence and Statistics. Fort Lauderdale, USA, 2017: 1273−1282.

[2] WANG Hongyi, YUROCHKIN M, SUN Yuekai, et al. Federated learning with matched averaging [EB/OL]. (2020−02−25)[2021−03−09]https://arxiv: 2002.06440, 2020.

[3] KOPPARAPU K, LIN E, ZHAO J. FedCD: Improving performance in non-IID federated learning [EB/OL]. (2020−07−27) [2021−03−09]https:// arxiv: 2006.09637, 2020.

[4] YU Hao, YANG Sen, ZHU Shenghuo. Parallel restarted SGD with faster convergence and less communication: Demystifying why model averaging works for deep learning[C]//Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence. Palo Alto, USA, 2019: 5693-5700.

[5] WANG Shiqiang, TUOR T, SALONIDIS T, et al. Adaptive federated learning in resource constrained edge computing systems[J]. IEEE journal on selected areas in communications, 2019, 37(6): 1205–1221.

[6] YU Hao, JIN Rong, YANG Sen. On the linear speedup analysis of communication efficient momentum SGD for distributed non-convex optimization[C]//Proceedings of the 36th International Conference on Machine Learning. Long Beach, USA, 2019: 7184−7193.

[7] JEONG E, OH S, KIM H, et al. Communication-efficient on-device machine learning: federated distillation and augmentation under Non-IID private data [EB/OL]. (2018−11−28)[2021−03−09]https:// arxiv: 1811.11479, 2018.

[8] HUANG Li, YIN Yifeng, FU Zeng, et al. LoAdaBoost: loss-based AdaBoost federated machine learning with reduced computational complexity on IID and non-IID intensive care data[J]. PLoS one, 2020, 15(4): e0230706.

[9] REDDI S, CHARLES Z, ZAHEER M, et al. Adaptive federated optimization [EB/OL]. (2021−09−08) [2021−10−09]https:// arXiv: 2003.00295, 2021.

[10] YANG Kai, FAN Tao, CHEN Tianjian, et al. A quasi-newton method based vertical federated learning framework for logistic regression[EB/OL]. (2019−12−04) [2021−09−08]https:// arXiv: 1912.00513, 2019.

[11] DHAKAL S, PRAKASH S, YONA Y, et al. Coded federated learning[C]//2019 IEEE Globecom Workshops (GC Wkshps). Waikoloa, USA, 2019: 1−6.

[12] WANG Cong, YANG Yuanyuan, ZHOU Pengzhan. Towards efficient scheduling of federated mobile devices under computational and statistical heterogeneity[J]. IEEE transactions on parallel and distributed systems, 2021, 32(2): 394–410.

[13] MALINOVSKIY G, KOVALEV D, GASANOV E, et al. From local SGD to local fixed-point methods for federated learning[C]//Proceedings of the 37th International Conference on Machine Learning. New York, USA, 2020: 6692−6701.

[14] HANZELY F, RICHTÁRIK P. Federated learning of a mixture of global and local models [EB/OL]. (2020−02−10)[2021−03−09]https:// arXiv: 2002.05516, 2020.

[15] ROTHCHILD D, PANDA A, ULLAH E, et al. FetchSGD: Communication-efficient federated learning with sketching[C]//Proceedings of the 37th International Conference on Machine Learning. New York, USA, 2020: 8253−8265.

[16] WANG Jialei, WANG Weiran, SREBRO N. Memory and communication efficient distributed stochastic optimization with minibatch-prox[C]//Proceedings of the 2017 Conference on Learning Theory. New York, USA, 2017: 1882−1919.

[17] LI Tian, HU Shengyuan, BEIRAMI A, et al. Federated multi-task learning for competing constraints[EB/OL]. [2021−03−09]https://openreview.net/forum?id=1ZN5y4yx6T1.

[18] LI Tian, SAHU A, ZAHEER M, et al. Federated optimization in heterogeneous networks[J]. Proceeding of machine learning and systems, 2020, 2: 429–450.

[19] ZHOU Pan, YUAN Xiaotong, XU Huan, et al. Efficient meta learning via minibatch proximal update[EB/OL]. (2019−12−08)[2021−03−09]https://openreview.net/forum?id=B1gSHVrx8S.

[20] PHONG L T, AONO Y, HAYASHI T, et al. Privacy-preserving deep learning via additively homomorphic encryption[J]. IEEE transactions on information forensics and security, 2018, 13(5): 1333–1345.

[21] GO A, BHAYANI R, HUANG Lei. Twitter sentiment classification using distant supervision[J]. CS224N project report, Stanford, 2009, 1(12): 2009.

[22] LECUN Y, BOTTOU L, BENGIO Y, et al. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11): 2278–2324.

[23] COHEN G, AFSHAR S, TAPSON J, et al. EMNIST: extending MNIST to handwritten letters[C]//2017 International Joint Conference on Neural Networks (IJCNN). Anchorage, USA, 2017: 2921−2926.

[24] Tung K K. Topics in Mathematical Modeling[M]. Princeton University Press, 2007.

[25] BALLES, LUKAS, PHILIPP HENNING. Dissecting adam: the sign, magnitude and variance of stochastic gradients[C]//International Conference on Machine Learning. PMLR, 2018: 404−413.

作者简介:

窦勇敢, 硕士研究生, 主要研究方向为联邦学习、语义分割。

袁晓彤, 教授, 博士生导师, 中国计算机学会计算机视觉专委会委员, 中国自动化学会模式识别与机器智能专委会委员, IEEE 会员, 主要研究方向为机器学习和计算机视觉。入选江苏省双创人才。发表学术论文 80 余篇。

关于提名 2022 年度中国人工智能学会会士候选人的通知

根据《中国人工智能学会会士评定工作办法》的规定, 2022 年度中国人工智能学会会士候选人提名工作即日启动。现将本会会士候选人提名要求及安排通知如下:

一、会士候选人资格

会士候选人必须同时满足以下条件:

(一)在人工智能领域的科学研究与技术开发方面做出国内外瞩目的突出成就及创新贡献。

(二)所做出的科研成果在人工智能相关产业起了引领作用, 或发表在高水平期刊或会议上的论文在国内外产生了广泛的学术影响。

(三)具有五年以上(对学会有突出贡献者不限)高级会员会龄, 并一直关心支持学会工作, 为学会建设做出了突出贡献。

二、会士候选人的提名

(一)会士评选采用提名制, 会士候选人须由被提名人之外的其他人提名产生。每位会士候选人须得到三名提名人的提名。

(二)学会会士是有效提名人, 每位提名人每年作为提名人提名会士候选人不得超过两人。

(三)会士提名人必须填写《中国人工智能学会会士候选人提名表》(见附件)。

三、会士评定程序

会士评定工作委员会对会士被提名人的资格与提名程序进行审核, 若符合要求, 则提名成立, 被提名人确定为正式候选人, 并将材料提交会士评定专家委员会依据《中国人工智能学会会士评定工作办法》评定遴选。会士评选结果将在学会网站上公示五个工作日; 若无异议, 正式当选为中国人工智能学会会士。

四、材料报送

请提名人填写《中国人工智能学会会士候选人提名表》并签字, 于 2022 年 5 月 31 日前发送至会士评定工作委员会办公室, 邮箱: zhb@caai.cn。

提名工作期间有问题请咨询会士评定工作委员会办公室。

联系人: 贾晓丽

电　话: 010-82686686

邮　箱: zhb@caai.cn

中国人工智能学会

2022 年 5 月 5 日